DOCUMENTATION NOTE

NPS Reality v15.1
Online Documentation (Revision 7)

This Documentation Note summarises the changed topics in the Revision 7 release of the NPS Reality v15.1 Online Documentation. For more detailed information, refer to the Online Documentation itself.

Northgate Public Services

The Online Documentation has been updated to refer to Northgate Public Services (UK) Limited, Northgate Public Services, or NPS (according to the context), and to NPS Reality (formally) or Reality (informally).

Reality on Unix Installation & Upgrade Guide

Several changes have been made to the Online Documentation topics that comprise the *Reality on Unix Installation & Upgrade Guide*.

Upgrading from an earlier release

In the *Pre-Upgrade* section, replace existing Steps 1 and 2 with:

1. Make sure users are logged off and prevented logging into all databases while administration work is carried out. You can use **lockdbase** for each database at host command level or **INHIBIT-LOGONS** from within each database. Only when the upgrade is complete and you want users to login in again should you enable logins.

2. Install the latest Reality and UNIX-Connect updates for the release you are currently running; …

Reality on Windows Installation & Upgrade Guide

Several changes have been made to the Online Documentation topics that comprise the *Reality on Windows Installation & Upgrade Guide*.

Installation

The following initial section has been added:

Running Setup.exe in Administrator mode

Before installing Reality, you must close all other applications, including any anti-virus software. You must also shut down the Reality remote tape server if you have this installed (run the command **Realrts -e**).

Due to security changes in Windows you must have full administrator privileges and use either a mounted virtual ISO image of the Reality software, or a physical DVD produced from the image and run **setup.exe** in administrator mode. If autorun is configured and you see the **Reality Solutions** screen, please exit this and run the **setup.exe** command from the root directory of the DVD in administrator mode.

Note: You must select the **setup.exe** file with a right-click and then use the **Run as administrator** option; a Windows popup will request confirmation.

Please do not use a file structure of a Reality delivery that has been copied to a remote or the local system, use only a physical or mounted DVD/ISO image.

Upgrading from an earlier release

In the *Pre-Upgrade* section, replace existing Steps 1 and 2 with:

1   Make sure users are logged off and prevented logging into all databases while administration work is carried out. You can use **lockdbase** for each database at host command level or **INHIBIT-LOGONS** from within each database. Only when the upgrade is complete and you want users to login in again should you enable logins.

2.   Install the latest Reality updates for the release you are currently running; …

Installing Updates

The initial paragraphs of this section have been replaced by the following:

Updates to Reality are made available on the NPS Reality Latest Updates webpage. These are normally supplied as a service pack containing the latest updates that are required to be loaded for any support requests to be progressed.

Please read the documents *Description of Recommended Updates* and *Installation Info File* on the webpage for details of the contents of the updates and any additional configuration that might be necessary.

To download the latest service pack, on the **Reality V15.1** tab, click the required **Windows 64bit Update** or **Windows 32bit Update** file and save the file with a right-click.

Note: To install a service pack, you must have at least 200 Mb available on the drive containing the `realman` folder.

Other support information is available on the NPS Reality Support webpage.

Caution
Before you install an update, ensure that you have an up-to-date backup of your existing data.

The initial steps of the *Installation Procedure* section have been updated as follows:

1. For each Reality database to be updated:

    - Save to elsewhere in the database any system file items that you have customised. Files that might have been customised include SYSPROG-PL, PROCLIB, BP, SYSBP, SYSBP.MSGS, SYSPL, SYS.BASLIB and BASIC-COMPILERS.

    - Ensure that all users are logged off.

    - Carry out **FILE-SAVE** and **VERIFY-SAVE**.

2. Log on to Windows; you must have full Windows administrator privileges.

3. Ensure that no users are logged into Reality. If necessary use the **LOGOFF** command.

4. From the Reality website, download the required service pack.

    The service pack's file name includes the Reality version number, the service pack number, and an indication of whether it is 32 bit or 64 bit. It will also have a **.rltyfx** file extension.

5. Use Windows Explorer to find the downloaded service pack, then right-click the file and select **Run as administrator** to begin the installation.

    The **.rltyfx** file extension ensures that service packs are automatically opened with the **install_fix** utility.

    Alternatively, running **install_fix** from the command prompt, again as administrator, allows additional control over the installation — but this should only be done under the direction of Reality Support personnel to resolve local issues.

6. Follow the on-screen prompts.

DataBasic debugger enhancements

The following new and modified debugger commands are documented:

?	Displays help pages.
?P	Displays the name of the program currently running.
B	Adds an entry to the break point table.
CP	Toggles cursor positioning.
D	Displays the current state of DataBasic debugger controls and options, as set by various debugger commands.
MA	Causes the debugger to be entered on all active statement lines in a named code module, by adding the module to the list of monitored modules.
MD	Removes monitoring for a named code module, or all code modules, by deleting them from the list of monitored modules.
ME	Causes the debugger to be entered on entry to the start of a named code module, by adding the module to the list of monitored modules.
MO	Causes the debugger to be entered on reaching the line that immediately follows the current call to another module.
MR	Causes the debugger to be entered on (re-)entry to a named code module from any other module (that is, after a CALL or RETURN command) by adding the module to the list of monitored modules.
MX	Causes the debugger to be entered immediately on returning from the current module to the calling module.

The **MA**, **ME**, **MR** and **MD** commands are a separately licensed feature of NPS Reality.

Miscellaneous changes

BREAK-KEY-ON and BREAK-KEY-OFF commands

Corrected to show that the *portNumber* parameter must be preceded by an open parenthesis:

BREAK-KEY-ON (*portNumber*
BREAK-KEY-OFF (*portNumber*

LIST-$STAT-FILE* command

Added a link to the **LIST-FILE-STATS** command.

LIST-FILE-STATS command

Corrected the first paragraph, as follows:

Prints a file statistics report from the current statistics stored in the STAT-FILE (default data section). This Proc does not generate new statistics, nor does it attempt to fix GFEs.

Added a link to the **LIST-$STAT-FILE*** command.

OPTIMUM-FRAME-SIZE, OPTIMUM-MODULO and SYS-UPDATE commands

In each case, corrected the *Restrictions* section to begin:

Only available in the SYSMAN account …

CREATE-FILE (TCL)

Updated *Syntax Elements* to make it clearer that if neither *m1* (dictionary modulo) nor *m2* (data modulo) are provided, automatic file sizing is assumed

DIR-VIEW (TCL)

Corrected to say that host files larger than 2 Gb are visible, but not readable, and can be processed by OPENSEQ:

Because the maximum size of a Reality item is 2 Gb, host files larger than this will be visible but will not be readable. DataBasic's **OPENSEQ** statement can be used to process large files by opening them directly.

ACCEPT and CONNECT statements (DataBasic)

Updated the TCP/IP accept and connect strings to include a new **mstimeout** parameter, as follows.

For **ACCEPT**:

*TCP*host**;port**=port{;**mstimeout**=milliseconds}{;option}...

where:

milliseconds	is a timeout in milliseconds which overrides the TIMEOUT specified in the ACCEPT statement (or specifies a timeout if the TIMEOUT clause is omitted).

For **CONNECT**:

*TCP*host**;port**=port{;**true_host**=true-host{;**true_port**=true-port}{;**proxy_user**=user:password}}{;**mstimeout**=milliseconds}{;**TLS**=[none || server]}{;option}...

where:

milliseconds	is a timeout in milliseconds which overrides the TIMEOUT specified in the CONNECT statement (or specifies a timeout if the TIMEOUT clause is omitted).

BASIC command (DataBasic)

Added option **I**, as follows:

I	Generates an implicit list of item-ids for items that generate compilation errors.

EQUATE statement (DataBasic)

Updated the explanation of the *litValue* parameter as follows:

litValue	A DataBasic expression enclosed in single quotes, double quotes or backslashes. This is similar to *value*, but can contain spaces and multiple statements separated by semi-colons.

Added the following comment:

An EQUATE is terminated by an EOL character, comma, or semi-colon; multi-line EQUATEs are not permitted

INPUT statement (DataBasic)

Amended the **Syntax Elements** section to make it clearer, as follows:

length	The maximum number of characters to be entered, after which the input is automatically completed and a RETURN is executed, unless an underscore … is also used. This is useful when programming fixed length input fields, as no delimiter needs to be entered. The maximum, and default, length is 240 characters.

PERFORM statement (DataBasic)

Corrected minor formatting issues.

SYSTEM function (DataBasic)

Added new Reality parameters, as follows:

Parameter	Return Value
116	Returns the call stack. SYSTEM(116) returns a multi attribute string with the first attribute always containing the current module name and the line number. If called from the main program (that is, with nothing on the stack) it also returns a second attribute consisting of the string Stack empty. If called with within a subroutine it returns an additional attribute for each level of call stack, consisting of a single type character followed by the calling module name and the line number. The type character is: C for an external CALL, G for a local GOSUB, or F for an external user function call.
117	Returns the program source line number.
118	Returns the program line number.

SYSTEM and REAL.SYSTEM functions (MultiValue Compatibility)

Added a new mvBase parameter, as follows:

Parameter	Return Value	Reality Equivalent
35	Returns the program source line number.	117

VARVALTYPE function (DataBasic)

Updated the type characters returned by VARVALTYPE, as follows:

00	Cleared, unassigned variable.
01	Scaled binary number.
02	Short string.
0401	Session handle.
0402	External object handle.
0404	File handle.
0405	Selection list handle.
0406	Index handle.
81	String number.
82	Indirect string.
U	Undefined.

Corrected the definition of "string number":

A string number is a number that has more digits than can be held in the 48-bit accumulator, excluding the sign bit. In other words, a number greater than 2^{47}-1 (140,737,488,355,327)

RHTTP_GET function (Reality Web Services)

Amended *Syntax Elements* to stress that "an empty string" does not mean a literal, as follows:

request   Either an initialised HTTP request (see **RHTTP_INIT_REQUEST**) or a variable containing an empty string.

Database Configuration Parameters

Added **NpuDelay** and **NpuRetries**, as follows:

NpuDelay

The maximum number of times that the **npu** printer utility will delay before each retry to connect to your network printer.

This should have a value between 1 and 60.

NpuRetries

The maximum number of times that the **npu** printer utility will retry to connect to your network printer.

If set to -1 it will retry forever.

Corrected **ShareSize** to say that the maximum memory size (in bytes) calculated by the example LIST statement should be divided by 10 KB (not 10) to arrive at a realistic value for the ShareSize parameter.

MultiValue Compatibility

(a) Updated **FLOPPYTOTAPE** with a new option, as follows:

S{*,n*}   Splits the tape image at a maximum size. If **S** is used on its own, a default size of 100 megabytes is assumed. Otherwise a maximum size can be specified, in megabytes; for example, **S,200** would specify a maximum size of 200 megabytes. Note the mandatory comma.

and added an appropriate example:

```
FLOPPYTOTAPE C:\images\tapes PFDATA (S,50
```

Converts the file **C:\images\tapes\PFDATA** into multiple Reality tape images, as required, with each file being 50 megabytes or less:

C:\images\tapes\PFDATA1.rci
C:\images\tapes\PFDATA2.rci
C:\images\tapes\PFDATA3.rci and so on ...

(b) Updated TAPETOFLOPPY with a new option, as follows:

U   Unites split images produced by **FLOPPYTOTAPE** with the **S** option.

and added an appropriate example:

```
TAPETOFLOPPY C:\images\tapes PFDATA.rci (U
TAPETOFLOPPY C:\images\tapes PFDATA?.rci
```

Either of these commands converts the Reality tape images **C:\images\tapes\PFDATA1.rci**, **PFDATA2.rci**, **PFDATA3.rci**, and so on into a MultiValue pseudo-floppy image. The image is saved in the file **C:\images\tapes\PFDATA**.

The Debugger

Added a link to *Summary of Debugger Commands*.

RealWeb API Text Structure routines

Corrected **RW-START_PARA** to **RW_START_PARA**.

Summary of Debugger Commands (DataBasic Debugger)

Corrected links to **N** and **X** options.

Class RSC (Reality Interface Classes)

Updated the example to use the recommended form of the constructor:

```
RSC sub = new RSC(con, "subname");
```

A Code - Algebraic Functions (Conversion Codes)

Amended description of the ND operand to include a note:

ND   Number of items (detail lines) since the last control break.

> *Note:*   On detail lines, this has a value of 1; on break lines, it is the number of detail lines since the last controlled-break; and on grand-total lines, it has the same value as the item counter.